REMARKS

Applicant is in receipt of the Office Action mailed March 17, 2008. Claims 1-18, and 26-28 are pending in the case. Reconsideration of the present case is earnestly requested in light of the following remarks.

Objections

Figures 5 and 8 were objected to as being too small. Figures 12 and 14 were objected to as being too dark.

Applicant has amended the Figures accordingly, and has further amended Figure 13 (on the same sheet as Figure 12) and Figure 15 (on the same sheet as Figure 14) per standard PTO drawing requirements.

Applicant thus requests removal of the objections to the Figures.

The Examiner objected to the Specification and requested that trademarked terms such as JAVA, etc., be capitalized. Applicant has amended the Specification accordingly, and requests removal of the objection to the Specification.

The Examiner further noted that erroneous page/line numbers for an amendment to the Specification were provided in the previous Response, which Applicant has corrected in the above amendments.

Section 112 Rejections

Claims 1-18 and 26 were rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement, specifically, for claiming "wherein each graphical program node comprises an icon and program code,…wherein the plurality of graphical program nodes comprises: a first plurality of function nodes …, and a second plurality of property nodes…[*sic*]". The Office Action asserts that the Specification supports the notion of function nodes with icons and underlying program code, but not property nodes, quoting the following definition from the present Specification:

> **Node** – In the context of a graphical program, an element that may be included in a graphical program. A node may have an associated icon that represents the node in the graphical program, as well as underlying code or data that implements functionality of the node. Exemplary nodes include function nodes, terminal nodes, structure nodes, etc. Nodes may be connected together in a graphical program by connection icons or wires.

As may be seen, this definition is generic to any kind of graphical program node– note that the list of exemplary nodes clearly does not limit graphical program nodes to function nodes, terminal nodes, and structure nodes, for at least the reasons that a) the list is exemplary, and 2) the use of "etc." at the end of the list. This point is made explicitly in p.26, lines 11-16, which states:

> …the plurality of graphical program nodes (or node icons) are organized in the display window in a hierarchy of graphical program nodes, where the hierarchy includes a first plurality of function nodes displayed in the display window, each corresponding to a respective functionality, and a second plurality of property nodes displayed in the display window, each corresponding to a respective one of at least a subset of the plurality of function nodes.

Clearly, the property nodes are included in the plurality of graphical program nodes, and thus the definition quoted above applies to property nodes as well as function nodes (and any other kind of graphical program node).

Applicant thus respectfully submits that support for property nodes including icons and program code is supported in the Specification, and so requests removal of the section 112 rejection of claims 1-18 and 26.

Section 102 Rejections

Claims 1-18 and 26-28 were rejected under 35 U.S.C. 102(b) as being anticipated by "LabVIEW Advanced Application Development, Part: Coding Techniques", published February 2002, henceforth, "LabVIEW".

Claim 1 recites:

1. A computer accessible memory medium comprising program instructions, wherein the program instructions are executable by a processor to implement:

displaying a palette, including a display window comprising a plurality of graphical program nodes for use in a graphical program, wherein each graphical program node comprises an icon and program code, wherein each graphical program node is represented by the graphical program node's respective icon in the palette and is selectable from the palette for inclusion in the graphical program;

wherein the plurality of graphical program nodes comprises:

a first plurality of function nodes displayed in the display window, wherein each function node corresponds to a respective functionality; and

a second plurality of property nodes displayed in the display window, wherein each property node corresponds to a respective one of at least a subset of the plurality of function nodes, wherein each property node is displayed proximate to said respective one of the at least a subset of the plurality of function nodes.

Nowhere does the cited reference disclose **wherein the plurality of graphical program nodes comprises: a first plurality of function nodes displayed in the display window, wherein each function node corresponds to a respective functionality; and a second plurality of property nodes displayed in the display window, wherein each property node corresponds to a respective one of at least a subset of the plurality of function nodes, wherein each property node is displayed proximate to said respective one of the at least a subset of the plurality of function nodes**, as recited in claim 1.

In asserting that LabVIEW discloses these features, the Office Action cites pp.32 and 34. Applicant respectfully notes that the LabVIEW reference is a (Power Point) slide show presentation regarding application development using the LabVIEW graphical program development environment, and generally describes various features and techniques available for users in development of such applications. Applicant respectfully notes that each page of the reference includes a presentation slide and notes for the presenter that include demonstration instructions for illustrating the features or techniques.

Page 32 describes an application control palette that includes a variety of graphical program nodes, specifically, an Open Application Reference node, an Open VI

Reference node, a Close LV Object Reference node, a Call By Reference node, a Property node, and an Invoke node. Page 34 is directed to Control Property nodes, and illustrates a block diagram (VI) and corresponding front panel (GUI) that includes a control for scrolling arrays using a slider control. The cited Control Property nodes presumably may be included in the block diagram and used to set properties of the control.

Applicant respectfully submits that neither of these citations discloses each property node corresponding to a respective function node, and being displayed proximate to the respective function node in the palette. For example, page 34 doesn't show a palette at all, and the Control Property nodes discussed on that page are for setting properties of controls, not function nodes, i.e., the Control Property nodes correspond to GUI elements, not function nodes. Page 32 illustrates a palette with various creation nodes and a property node; however, Applicant respectfully notes that the property node shown on page 32 is for setting attributes of objects, e.g., an application, VI, or front panel object, e.g., a GUI control or indicator, and is nowhere described as corresponding to a respective function node, nor is the cited property node displayed proximate to a corresponding function node. For example, note that the cited property node, which is generic until wired to receive an object, (after being wired so) would actually correspond to the object provided to the property node, and specifically does *not* correspond to any of the various other nodes shown in the palette, i.e., the Open Application Reference node, Open VI Reference node, Close LV Object Reference node, Call By Reference node, or Invoke node. Examples of function-specific property nodes are described in p.28:2-25, including a channel property node, a timing property node, a trigger property node, a read property node, and a write property node, where the channel property node corresponds to the task or channel creation nodes, e.g., task or channel configuration, the timing property node corresponds to the timing node, the trigger property node corresponds to the trigger node, the read property node corresponds to the read node, and the write property node corresponds to the write node. This is in direct contrast to the generic property node shown on page 32 of LabView.

The display of function-specific property nodes proximate to their corresponding function nodes is described on p.28, which indicates that each property node is displayed in a manner that indicates its respective association or correspondence with a function node, e.g., the timing property node is presented in the same column as the timing node, the read property node is presented in the same column as the read node, and so forth. Regarding non-function-specific property nodes such as that shown in LabView p.32, this text states that property nodes that do not specifically correspond to particular function nodes may be displayed separately from those that do, e.g., not proximate to the function nodes.

Thus, Applicant submits that LabVIEW fails to teach or suggest these claimed features, and so claim 1 and those claims respectively dependent therefrom are patentably distinct and non-obvious over the cited art, and are thus allowable.

Independent claims 27 and 28 include similar limitations as claim 1, and so the above arguments apply with equal force to these claims. Thus, for at least the reasons provided above, Applicant submits that claims 27 and 28, and those claims respectively dependent therefrom, are patentably distinct and non-obvious over the cited art, and are thus allowable.

Applicant also asserts that numerous ones of the dependent claims recite further distinctions over the cited art.

For example, LabVIEW fails to disclose **wherein the first plurality of function nodes comprises two or more of: a channel creation node; a read node; or a write node**, as recited in claim 5.

Cited p.26 discloses (two) displays of various function nodes, including read and write VIs; however, Applicant respectfully notes that the cited displays do not include function-specific property nodes, and so, since the claimed first plurality of nodes is displayed in a palette with corresponding function-specific property nodes, the p.26 citation fails to teach or suggest this feature.

Moreover, p.55 is directed to graphical object-oriented techniques, and neither illustrates, nor makes mention of displaying any of these function nodes as claimed.

Thus, the cited art fails to disclose these features of claim 5, and so claim 5 and those claims respectively dependent therefrom are patentably distinct and non-obvious over the cited art, and are thus allowable.

As another example, LabVIEW fails to disclose **wherein the first plurality of function nodes further comprises: a wait until done node**, as recited in claim 6.

Cited p.13 neither illustrates nor mentions a wait until done node, nor displaying such a node as claimed. P.16 illustrates and describes an example state-machine that uses a while loop with shift registers, but fails to disclose a wait until done node, and more specifically, displaying such a wait until done node as claimed.

Thus, the cited art fails to disclose these features of claim 6, and so claim 6 and those claims respectively dependent therefrom are patentably distinct and non-obvious over the cited art, and are thus allowable.

As yet another example, LabView fails to disclose **wherein the first plurality of function nodes further comprises one or more of: a timing node; a triggering node; a start node; a stop node; or a clear node**, as recited in claim 8.

The cited stop and exit nodes shown in p.32 are not a timing node, a triggering node, a start node, a stop node, or a clear node, and so this citation fails to teach or suggest this feature, and so claim 8 and those claims respectively dependent therefrom are patentably distinct and non-obvious over the cited art, and are thus allowable.

As a further example, Applicant submits that LabView fails to disclose **wherein, in each property node being displayed proximate to the respective one of the at least a subset of the plurality of function nodes, each property node is displayed in one of: a common row with the respective one of the at least a subset of the plurality of function nodes; or a common column with the respective one of the at least a subset of the plurality of function nodes**, as recited in claim 14.

Cited p.57 is directed to graphical object-oriented programming (OOP), and particularly to synchronization using Semaphore VIs. However, the synchronization palette disclosed therein in no way teaches or suggests function-specific property nodes being displayed in the palette in a common row or column with respective corresponding function nodes. In fact, no function-specific property nodes are disclosed at all, nor any other kind of property node.

Cited p.61 is directed to a graphical OOP, and mentions adding a Chronometer Class to a palette, but nowhere mentions or even hints at the particular arrangement of function nodes and function-specific property nodes claimed.

Thus, these citations fail to teach or suggest these features of claim 14, and so claim 14 and those claims respectively dependent therefrom are patentably distinct and non-obvious over the cited art, and are thus allowable.

Moreover, Applicant has reviewed the rest of the reference closely, and can find no teaching or suggestion of function-specific property nodes and corresponding function nodes being displayed in a palette at all.

Applicant also asserts that numerous other ones of the dependent claims recite further distinctions over the cited art. However, since the independent claims have been shown to be patentably distinct, a further discussion of the dependent claims is not necessary at this time.

Removal of the section 102 rejection of claims 1-18 and 26-28 is earnestly requested.

CONCLUSION

Applicant submits the application is in condition for allowance, and an early notice to that effect is requested.

If any extensions of time (under 37 C.F.R. § 1.136) are necessary to prevent the above-referenced application(s) from becoming abandoned, Applicant(s) hereby petition for such extensions. The Commissioner is hereby authorized to charge any fees which may be required or credit any overpayment to Meyertons, Hood, Kivlin, Kowert & Goetzel P.C., Deposit Account No. 50-1505/5150-81100/JCH.

Also filed herewith are the following items:

☒ Replacement Sheets for Figures 5, 8, 12, and 14

□ Terminal Disclaimer

□ Power of Attorney By Assignee and Revocation of Previous Powers

□ Notice of Change of Address

□ Other:

Respectfully submitted,

/Jeffrey C. Hood/

Jeffrey C. Hood, Reg. #35198
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert & Goetzel PC
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8800
Date: 2008-06-09 JCH/MSW